

Interner Verteilerschlüssel:

- (A) [-] Veröffentlichung im ABl.
- (B) [-] An Vorsitzende und Mitglieder
- (C) [-] An Vorsitzende
- (D) [X] Keine Verteilung

**Datenblatt zur Entscheidung
vom 13. Januar 2017**

Beschwerde-Aktenzeichen: T 2468/10 - 3.5.07

Anmeldenummer: 03101382.4

Veröffentlichungsnummer: 1369790

IPC: G06F17/30

Verfahrenssprache: DE

Bezeichnung der Erfindung:

Verfahren zur dynamischen Generierung strukturierter Dokumente

Anmelder:

Siemens Aktiengesellschaft

Stichwort:

Dynamische Dokumentengenerierung/SIEMENS

Relevante Rechtsnormen:

EPÜ Art. 84

VOBK Art. 13(1), 13(3)

Schlagwort:

Spät eingereichter Antrag - Haupt- und Hilfsantrag
(zugelassen)

Patentansprüche - Stützung durch die Beschreibung - Haupt- und
Hilfsantrag (nein)

Zitierte Entscheidungen:

T 1942/12, T 2068/14

Orientierungssatz:



Beschwerdekammern
Boards of Appeal
Chambres de recours

European Patent Office
D-80298 MUNICH
GERMANY
Tel. +49 (0) 89 2399-0
Fax +49 (0) 89 2399-4465

Beschwerde-Aktenzeichen: T 2468/10 - 3.5.07

E N T S C H E I D U N G
der Technischen Beschwerdekammer 3.5.07
vom 13. Januar 2017

Beschwerdeführer: Siemens Aktiengesellschaft
(Anmelder) Wittelsbacherplatz 2
80333 München (DE)

Vertreter: Siemens AG
Postfach 22 16 34
80506 München (DE)

Angefochtene Entscheidung: Entscheidung der Prüfungsabteilung des Europäischen Patentamts, die am 20. Juli 2010 zur Post gegeben wurde und mit der die europäische Patentanmeldung Nr. 03101382.4 aufgrund des Artikels 97 (2) EPÜ zurückgewiesen worden ist.

Zusammensetzung der Kammer:

Vorsitzender M. Rognoni
Mitglieder: R. de Man
R. Cramer

Sachverhalt und Anträge

I. Die Anmelderin (Beschwerdeführerin) hat gegen die Entscheidung der Prüfungsabteilung, die europäische Patentanmeldung Nr. 03101382.4 zurückzuweisen, Beschwerde eingelegt.

II. In der Entscheidung wurden u.a. folgende Dokumente zitiert:

D6: O'Brien M.: "Embedded Web Servers", Embedded Systems Programming, November 1999, Seiten 67-72;

D7: "Comparing JavaServer Pages Technology and Microsoft Active Server Pages - An Analysis of Functionality", 1. November 1999, Seiten 1-8, gefunden im Internet: http://java.sun.com/products/jsp/pdf/jsp_asp.pdf;

D8: Burchert F. et al.: "Remote-Administration von eingebetteten Systemen mit einem Java-basierten Add-On-Modell", Tagungsband Java-Informationen-Tage 1999; und

D9: Jazdi N.: "Component-based and Distributed Web Application for Embedded Systems", International Conference on Intelligent Agents, Web Technology and Internet Commerce, 2001.

Die Prüfungsabteilung kam zu dem Ergebnis, dass der Hauptantrag nicht die Erfordernisse des Artikels 123 (2) EPÜ, die Hilfsanträge 1, 2 und 3 im Hinblick auf die Dokumente D5 bis D9 nicht die Erfordernisse der Artikel 52 (1) und 56 EPÜ, und der Hilfsantrag 4 nicht die Erfordernisse des Artikels 123 (2) EPÜ erfüllten.

III. Mit der Beschwerdebegründung reichte die Beschwerdeführerin einen Hauptantrag und einen Hilfsantrag ein,

die auf den bisherigen Hilfsanträgen 2 und 3 basierten. Hilfsweise wurde die Durchführung einer mündlichen Verhandlung als Videokonferenz beantragt.

- IV. In einem der Ladung zur mündlichen Verhandlung am 13. Januar 2017 beigefügten Bescheid verwies die Kammer auf folgendes Dokument:

D10: "The J2EE Tutorial - The Life Cycle of a JSP Page", 1. November 2001, gefunden im Internet:
https://web.archive.org/web/20011101105938/http://java.sun.com/j2ee/tutorial/1_3-fcs/doc/JSPIntro4.html.

Sie teilte als ihre vorläufige Meinung mit, dass der Gegenstand des Anspruchs 1 beider Anträge im Hinblick auf das Dokument D6 nicht erfinderisch zu sein scheine. Zusätzlich wurden Einwände unter den Artikeln 84 und 123 (2) EPÜ erhoben. Der Antrag auf Durchführung der mündlichen Verhandlung als Videokonferenz wurde abgelehnt. Unter Verweis auf Artikel 13 (1) und (3) VOBK wurde hinzugefügt, dass schriftliche Eingaben oder Änderungen nicht später als einen Monat vor dem anberaumten Termin der mündlichen Verhandlung eingereicht werden sollten.

- V. Mit Schreiben vom 12. Januar 2017, das per Fax um 12:43 am selben Tag beim EPA einging, reichte die Beschwerdeführerin "zur Behebung der Bedenken der Kammer hinsichtlich Klarheit und ursprünglicher Offenbarung" einen neuen Haupt- und Hilfsantrag ein.

- VI. Die mündliche Verhandlung fand am 13. Januar 2017 statt. Am Ende der Verhandlung verkündete der Vorsitzende die Entscheidung der Kammer.

VII. Die Beschwerdeführerin beantragte, die angefochtene Entscheidung aufzuheben und auf der Grundlage des Hauptantrags oder des Hilfsantrags ein Patent zu erteilen.

VIII. Anspruch 1 des Hauptantrags lautet wie folgt:

"Verfahren zur dynamischen Generierung strukturierter Dokumente (SD) an mindestens einem, mit einem Client (CL) kommunizierenden, in seinen Ressourcen limitierten, mikrocontroller-basierten Leitreechner (SRV), der als Kommunikationseinrichtung zur Vermittlung von Kommunikationsendgeräten ausgebildet ist, mit

- zumindest einem einzigen Vorlagedokument (TD) mit enthaltenen Aufrufen von Dienstnehmern (JB), das
 - sowohl in dem in seinen Ressourcen limitierten, mikrocontroller-basierten Leitreechner als auch in einem Leitreechner mit ausreichenden Ressourcen herangezogen werden kann,
 - für den Leitreechner ausreichender Ressourcen entwickelt und nach abgeschlossener Entwicklung in gleicher Weise auf dem in seinen Ressourcen limitierten, mikrocontroller-basierten Leitreechner herangezogen wird,
 - in Form einer Java Serverpage vorliegt, wobei die Dienstnehmer in Form von durch eine Java Virtual Machine bearbeitbaren JavaBeans vorliegen,
- einer Laufzeitumgebung eines Kontrollmoduls (CRT) zur Umsetzung des Vorlagedokuments, die dazu nicht mit einer Java Virtual Machine in Verbindung mit einer Servlet-Engine gebildet, sondern durch in der Sprache C++ programmierte Softwareimplementierungen

im Kontrollmodul realisiert ist und im wesentlichen aus einem Parser und einem Scheduler besteht, wobei der Parser ohne eine Java Virtual Machine oder eine Servlet-Engine ausgestaltet ist,

- Leitrechnerkontrollfunktionen (SCF), die als architektur-, anwendungsprogramm- und/oder betriebssystemabhängige Befehle des Leitrechners ausgebildet sind,
- einem Schnittstellenmodul (IF) zum Zugriff auf die Leitrechnerkontrollfunktionen, bestehend aus einer anwendungsspezifischen, architekturabhängigen Software, die zumindest Datenbankzugriffe oder einen Zugriff auf Konfigurationsparameter des Leitrechners mit einem definierten Befehlssatz gestattet,

das Verfahren umfassend die Schritte:

- a) Empfang von Anforderungsdaten (REQ) des Clients am Leitrechner,
- b) Extraktion von Anfrageparametern aus den Anforderungsdaten (REQ),
- c) dynamische Generierung des strukturierten Dokuments unter Verwendung mindestens eines der Vorlagedokumente,
 - i) wobei die extrahierten Anfrageparameter auf einen korrespondierenden Befehlssatz des Schnittstellenmoduls abgebildet werden,
 - ii) wobei Anfrageparameter ignoriert werden, für die kein zugehöriger Befehl im Befehlssatz des Schnittstellenmoduls existiert,
 - iii) wobei die Java Serverpage durch die Laufzeitumgebung nicht mit einer Java Virtual Machine in Verbindung mit einer Servlet-Engine korrekt umgesetzt wird, sondern indem die in der Java Serverpage enthaltenen JavaBean-Aufrufe unter

Heranziehung der abgebildeten
Anfrageparameter in der Laufzeitumgebung
ausgeführt werden, wo sie durch den Parser
und den Scheduler aus der Java Serverpage
extrahiert werden und ein sehr kleiner
Ausschnitt der JavaBean-Syntax auf einen
korrespondierenden Befehlssatz des
Schnittstellenmoduls IF abgebildet wird,
und nach derart erfolgter Ausführung
Inhalte und/oder Struktur des
strukturierten Dokuments definieren,

- d) Übermittlung des dynamisch generierten
strukturierten Dokuments an den Client."

IX. Anspruch 1 des Hilfsantrags weist im Vergleich zum
Anspruch 1 des Hauptantrags folgende Unterschiede auf:

- nach dem Text "in Form einer Java Serverpage ...
vorliegen" ist Folgendes hinzugefügt worden:

"o auf einer beliebigen Rechneinheit,
insbesondere der des Clients, gespeichert ist,";
- im Schritt c) sind die Unterschritte i), ii) und
iii) in ii), iii) und iv) umnummeriert worden und
folgender Schritt ist hinzugefügt worden:

"i) wobei von dem Kontrollmodul auf die Daten des
Vorlagedokuments über ein paketorientiertes
Netzwerk (NW) sowie über eine Ein-Ausgabeeinheit
(IO) des Leitrechners zugegriffen wird,"; und
- im Schritt iv) ist der Text "nach derart erfolgter
Ausführung" durch den Text "nach derart in der
Laufzeitumgebung erfolgter Ausführung" ersetzt
worden.

- X. Die Beschreibung gemäß Haupt- und Hilfsantrag entspricht der ursprünglich eingereichten und in der A2-Veröffentlichung enthaltenen Beschreibung.
- XI. Entscheidungsrelevante Argumente der Beschwerdeführerin sind in den Entscheidungsgründen wiedergegeben.

Entscheidungsgründe

- 1. Die Beschwerde genügt den in den Artikeln 106 bis 108 und Regel 99 EPÜ genannten Bestimmungen und ist somit zulässig.
- 2. *Antrag auf Durchführung der mündlichen Verhandlung als Videokonferenz*

Da ein allgemeiner Rahmen für die Durchführung von mündlichen Verhandlungen vor den Beschwerdekammern als Videokonferenz zurzeit fehlt, werden entsprechende Anträge grundsätzlich abgelehnt (vgl. die Entscheidung T 1942/12 von 3. September 2015, Gründe 2). Im vorliegenden Fall kommt hinzu, dass die Beschwerdeführerin ihren Antrag nicht begründet hat, so dass für die Kammer nicht ersichtlich ist, aufgrund welcher außergewöhnlichen Umstände hiervon abzuweichen wäre (vgl. die Entscheidung T 2068/14, Gründe 1.2.5). Daher fand die mündliche Verhandlung auf herkömmliche Weise statt.

- 3. *Zulassung des Haupt- und Hilfsantrags*

Die neuen Anträge wurden am Nachmittag vor der mündlichen Verhandlung "zur Behebung der Bedenken der

Kammer hinsichtlich Klarheit und ursprünglicher Offenbarung" eingereicht.

Obwohl die Beschwerdeführerin grundsätzlich berechtigt war, die erstmals im Bescheid der Kammer unter den Artikeln 84 und 123 (2) EPÜ erhobenen Einwände mittels Änderungen ihrer Anträge zu beheben, wäre es angebracht gewesen, solche Änderungen entsprechend dem Hinweis der Kammer spätestens einen Monat vor dem anberaumten Termin der mündlichen Verhandlung einzureichen. Durch die Vorlage neuer Anträge zu einem Zeitpunkt, an dem die Kammer ihre Vorbereitung auf die mündliche Verhandlung schon weitgehend abgeschlossen hatte, hat die Beschwerdeführerin das Risiko von deren Nichtzulassung erheblich vergrößert.

Außerdem wurde während der mündlichen Verhandlung deutlich, dass die vorgenommenen Anspruchsänderungen nicht nur als Reaktion auf Einwände bezüglich Klarheit und ursprünglicher Offenbarung vorgesehen waren, sondern es mit diesen Änderungen auch beabsichtigt war, die Kammer von ihrer bisherigen Auslegung des beanspruchten Gegenstands abzubringen.

Da die Kammer dennoch zu der Überzeugung gelangt ist, ohne Verlegung der mündlichen Verhandlung über die neuen Anträge materiell entscheiden zu können, hat sie in Ausübung ihres Ermessens die neuen Anträge in das Verfahren zugelassen (Artikel 13 (1) und (3) EPÜ).

4. *Die Anmeldung*

- 4.1 Die Beschreibung der Anmeldung weist im Absatz [0012] der A2-Veröffentlichung auf bekannte Verfahren für die dynamische Generierung strukturierter Dokumente wie HTML-Dokumente an einem mit einem Client

kommunizierenden Leitrechner hin. Eines dieser Verfahren verwendet auf dem Leitrechner eine Laufzeitumgebung zur Ausführung von Skripten wie Active Server Pages, Java Server Pages oder von "echten" Skriptsprachen wie Perl, PHP und Python. Bei der Verarbeitung dieser Skripten wird ein Skriptinterpreter oder, im Fall von Java Server Pages, eine Java Virtual Machine (JVM) verwendet. Eine Java Server Page (JSP) stellt eine Mischung aus statischen HTML-Ausdrücken und dynamischen Objekt- oder JavaBean-Aufrufen dar. Laut Absatz [0012] können JavaBeans in Java Server Pages oder reinen serverseitig ausführbaren Servlets eingebunden werden. Die Java Server Page wird zunächst im Rahmen einer Servlet Engine serverseitig in ein Servlet umgesetzt. Wird im Folgenden durch einen Client die zu dieser Java Server Page korrespondierende URL durch einen externen HTML-Browser aufgerufen, ruft die Servlet Engine ihrerseits das "vorkompilierte" Servlet auf, das dann den in der Java Server Page definierten HTML-Inhalt unter Verwendung der eingebundenen JavaBeans dynamisch erzeugt.

4.2 Die Verwendung einer JVM in Verbindung mit einer Servlet Engine ist zu ressourcenaufwändig für bestimmte Systemkonfigurationen, wie z.B. einen in einer "Embedded System"-Architektur ausgestalteten und in seiner Arbeitsspeicherkapazität begrenzten Leitrechner. Ziel der Erfindung ist es, dieses Problem zu umgehen und eine Generierung strukturierter Dokumente auf Basis von Java Server Pages an einem ressourcenbegrenzten Leitrechner zu ermöglichen.

4.3 In den Absätzen [0039] bis [0041] der Beschreibung offenbart die Anmeldung die vorgeschlagene Lösung im Wesentlichen wie folgt.

In einem ressourcenbegrenzten Leitrechner wird die Laufzeitumgebung zur korrekten Umsetzung der JSP nicht mit einer extrem rechen- und speicherplatzaufwändigen JVM in Verbindung mit einer Servlet-Engine gebildet. Stattdessen wird die Laufzeitumgebung durch "schlankere" Softwareimplementierungen realisiert, die beispielsweise in der Sprache C++ programmiert sind. Im Wesentlichen besteht diese Laufzeitumgebung aus einem vereinfachten Parser und einem sogenannten "Scheduler".

Ein Parser ist ein Programm, das Textteile eines Dokuments auf Basis einer semantischen Analyse des Textes durch Befehle oder Code ersetzt. Ein Scheduler führt Funktionen einer Ablaufsteuerung durch.

Durch diese Laufzeitumgebung werden die im Skelett des Vorlagedokuments eingebetteten JavaBeans bzw. Aufrufe dieser JavaBeans durch den Parser und den Scheduler extrahiert und auf einen korrespondierenden Befehlssatz eines Schnittstellenmoduls abgebildet. Diese Abbildung kann sehr vereinfacht gestaltet sein, indem sie sich beispielsweise auf einen sehr kleinen Ausschnitt der JavaBean- bzw. JSP-Syntax beschränkt. Dies ermöglicht eine ebenso einfache und "schlanke" Ausgestaltung des Parsers ohne eine JVM oder eine Servlet-Engine auf einem Embedded System.

- 4.4 Außerdem offenbart der Absatz [0018], dass zunächst der Leitrechner von einem Client übermittelte Anforderungsdaten für eine Anforderung eines dynamisch zu gestaltenden strukturierten Dokuments empfängt und Anfrageparameter aus den Anforderungsdaten extrahiert. Diese Anfrageparameter werden ebenfalls "auf einen Befehlssatz eines architekturenspezifischen Schnittstellenmodul[s] am Leitrechner abgebildet", wobei die Anfrageparameter durch für das Schnittstellenmodul

verständliche Befehle ersetzt werden und fallweise Anfrageparameter ignoriert werden, für die kein zugehöriger Befehl im Befehlssatz des Schnittstellenmoduls existiert. Dann erzeugt der Leitreechner unter Verwendung eines Vorlagedokuments ein den Anfrageparametern entsprechendes strukturiertes Dokument. Das Vorlagedokument enthält einen oder mehrere Dienstnehmer, die unter Heranziehung der abgebildeten Anfrageparameter in einer vom Schnittstellenmodul zur Verfügung gestellten Laufzeitumgebung ausgeführt werden.

- 4.5 Der ursprüngliche Anspruch 1 korrespondiert im Wesentlichen mit Absatz [0018]. Der abhängige Anspruch 7 fügt hinzu, dass das Schnittstellenmodul einen Parser und einen Scheduler enthält, mit denen die Anweisungen der Dienstnehmer extrahiert und auf die architekturenspezifische Laufzeitumgebung des Leitrechners abgebildet werden.

5. *Hauptantrag - Stützung durch die Beschreibung*

- 5.1 Laut Anspruch 1 des Hauptantrags liegen das Vorlagedokument in Form einer Java Server Page und die Dienstnehmer in Form von durch eine Java Virtual Machine bearbeitbaren JavaBeans vor. Das Vorlagedokument enthält Aufrufe von Dienstnehmern, d.h. Aufrufe von JavaBeans.

Gemäß dem beanspruchten Verfahren werden zuerst Anfrageparameter aus den vom Client empfangenen Anforderungsdaten extrahiert und auf einen korrespondierenden Befehlssatz des Schnittstellenmoduls abgebildet, wobei Anfrageparameter ignoriert werden, für die kein zugehöriger Befehl im Befehlssatz des Schnittstellenmoduls existiert.

Dann wird die Java Server Page durch die Laufzeitumgebung "nicht mit einer Java Virtual Machine in Verbindung mit einer Servlet-Engine korrekt umgesetzt ... , sondern indem die in der Java Serverpage enthaltenen JavaBean-Aufrufe unter Heranziehung der abgebildeten Anfrageparameter in der Laufzeitumgebung ausgeführt werden". Hierbei werden die JavaBean-Aufrufe aus der Java Server Page extrahiert und "ein sehr kleiner Ausschnitt der JavaBean-Syntax wird auf einen korrespondierenden Befehlssatz des Schnittstellenmoduls IF abgebildet".

5.2 Laut Beschwerdeführerin ist Anspruch 1 des Hauptantrags so zu verstehen, dass weder das Vorlagedokument noch die Laufzeitumgebung JavaBeans, d.h. in der Java-Programmiersprache programmierte Softwareimplementierungen, enthält. Stattdessen würden die im Vorlagedokument enthaltenen JavaBean-Aufrufe auf Aufrufe von in der Sprache C++ programmierten Softwareimplementierungen abgebildet, die den Funktionalitäten der JavaBeans entsprechen. Die Laufzeitumgebung brauche deshalb überhaupt nicht über die Fähigkeit zu verfügen, Javacode auszuführen.

5.3 Die Kammer ist mit dieser Auslegung des Anspruchs einverstanden. Laut Anspruch 1 besteht die Laufzeitumgebung nicht aus einer Java Virtual Machine in Verbindung mit einer Servlet-Engine. Die Java Server Page wird korrekt umgesetzt, indem die darin enthaltenen JavaBean-Aufrufe extrahiert und in der Laufzeitumgebung ausgeführt werden. In diesem Zusammenhang ist es technisch sinnvoll, das Merkmal, dass "ein sehr kleiner Ausschnitt der JavaBean-Syntax auf einen korrespondierenden Befehlssatz des Schnittstellenmoduls IF abgebildet wird", im Einklang

mit den Erörterungen der Beschwerdeführerin so auszulegen, dass ein kleiner Ausschnitt aller möglichen JavaBean-Aufrufe auf Aufrufe von (schlankeren) in der Sprache C++ programmierten Softwareimplementierungen abgebildet wird.

- 5.4 Die Kammer ist jedoch aus den folgenden Gründen der Auffassung, dass der Anspruch nicht von der Beschreibung gestützt wird.

Laut Beschreibung kann ein Vorlagedokument nicht nur Aufrufe von JavaBeans oder, im Allgemeinen, von Dienstnehmern enthalten, sondern auch die JavaBeans oder Dienstnehmer selbst (siehe Absatz [0012], "Diese JavaBeans können wiederum in Java Server Pages oder reinen serverseitig ausführbaren sogenannten 'Servlets' eingebunden werden"; Absatz [0013], "Mit dem Begriff 'Dienstnehmer' ... wird auf Skripten oder Komponenten innerhalb eines ... HTML-Dokuments Bezug genommen"; Absatz [0018], "Das Vorlagedokument enthält einen oder mehrere Dienstnehmer"; Figur 2 in Verbindung mit Absatz [0034], "JavaBeans JB als weiteren Bestandteil des Vorlagedokuments TD"; Absatz [0041], "die im Skelett SQC des Vorlagedokuments eingebetteten JavaBeans JB bzw. Aufrufe dieser JavaBeans JB"). Absatz [0018] offenbart, dass erfindungsgemäß die im Vorlagedokument enthaltenen Dienstnehmer in der Laufzeitumgebung ausgeführt werden. Laut Absatz [0041] werden hierbei durch den Parser und den Scheduler der Laufzeitumgebung die im Vorlagedokument eingebetteten JavaBeans bzw. Aufrufe dieser JavaBeans extrahiert und auf einen Befehlssatz des Schnittstellenmoduls abgebildet. Absatz [0040] erläutert, dass der Parser mittels einer semantischen Analyse Textteile des Vorlagedokuments durch Befehle oder Code ersetzt.

Die Beschreibung offenbart deshalb, dass erfindungsgemäß die Dienstnehmer bzw. die JavaBeans in der Laufzeitumgebung ausgeführt werden. Hierzu werden die Dienstnehmer auf einen Befehlssatz "abgebildet". In diesem Zusammenhang ist nach Ansicht der Kammer im Absatz [0041] die Aussage "Diese Abbildung kann dabei sehr vereinfacht gestaltet sein, indem sie sich beispielsweise auf einen sehr kleinen Ausschnitt der JavaBean- bzw. JSP-Syntax beschränkt" so zu verstehen, dass die Laufzeitumgebung nur einen Ausschnitt der Syntax der Java-Programmiersprache (in der JavaBeans bekannterweise realisiert sind) unterstützt.

Die in der Beschreibung offenbarte Erfindung erfordert somit nicht, dass die in der Sprache C++ programmierten Softwareimplementierungen die Funktionalitäten eines Ausschnitts der durch das Vorlagedokument aufgerufenen JavaBeans implementieren. Stattdessen wird der Javacode der JavaBeans oder mindestens der unterstützte Teil dieses Javacodes ausgeführt. Da nur ein Teil der Standardfunktionalitäten unterstützt wird, braucht die Laufzeitumgebung im Einklang mit Absatz [0039] nicht mit einer extrem rechen- und speicherplatzaufwändigen JVM in Verbindung mit einer Servlet-Engine implementiert zu sein, sondern es reicht eine "schlankere" Implementierung aus.

In der Tat enthält die Beschreibung keinen Hinweis, dass die in der Sprache C++ programmierten Softwareimplementierungen Funktionalitäten von JavaBeans bereitstellen. Weiterhin bestätigt der ursprüngliche Anspruch 7, dass mit dem Parser und dem Scheduler der Laufzeitumgebung die Anweisungen der Dienstnehmer extrahiert und auf die Laufzeitumgebung abgebildet werden.

5.5 In der mündlichen Verhandlung hat die Beschwerdeführerin geltend gemacht, dass Java Server Pages gemäß den maßgeblichen Standards keine JavaBeans enthalten. Aber selbst wenn dies zutreffen würde (mit der Folge, dass die Beschreibung somit etliche Ungenauigkeiten enthält), würde sich an der technischen Lehre der Beschreibung, dass JavaBeans oder Dienstnehmer in der Laufzeitumgebung ausgeführt werden, nichts Wesentliches ändern.

5.6 Da die dem Anspruch 1 zugrundeliegende Lehre von der Beschreibung nicht gestützt wird, verstößt der Hauptantrag gegen die Erfordernisse von Artikel 84 EPÜ.

6. *Hilfsantrag - Stützung durch die Beschreibung*

Anspruch 1 des Hilfsantrags unterscheidet sich von Anspruch 1 des Hauptantrags nicht durch Merkmale, welche die in obigen Punkten 5.1 bis 5.5 dargelegte mangelnde Stützung des beanspruchten Gegenstands durch die Beschreibung überwinden könnten. Deshalb verstößt auch der Hilfsantrag gegen die Erfordernisse von Artikel 84 EPÜ.

7. *Schlussfolgerung*

Da keiner der Anträge der Beschwerdeführerin gewährbar ist, ist die Beschwerde zurückzuweisen.

Entscheidungsformel

Aus diesen Gründen wird entschieden:

Die Beschwerde wird zurückgewiesen.

Der Geschäftsstellenbeamte:

Der Vorsitzende:



G. Nachtigall

M. Rognoni

Entscheidung elektronisch als authentisch bestätigt